IMPROVING MANAGEMENT EFFECTIVENESS THROUGH DECISION SUPPORT SYSTEMS

INPUT

ABOUT INPUT

INPUT provides planning information, analysis, and recommendations to managers and executives in the information processing industries. Through market research, technology forecasting, and competitive analysis, INPUT supports client management in making informed decisions. Continuing services are provided to users and vendors of computers, communications, and office products and services.

The company carries out continuous and in-depth research. Working closely with clients on important issues, INPUT's staff members analyze and interpret the research data, then develop recommendations and innovative ideas to meet clients' needs. Clients receive reports, presentations, access to data on which analyses are based, and continuous consulting.

Many of INPUT's professional staff members have nearly 20 years experience in their areas of specialization. Most have held senior management positions in operations, marketing, or planning. This expertise enables INPUT to supply practical solutions to complex business problems.

Formed in 1974, INPUT has become a leading international consulting firm. Clients include over 100 of the world's largest and most technically advanced companies.

Headquarters
2471 East Bayshore Road
Suite 600
Palo Alto, California 9430
(415) 493-1600
Telex 171407

Detroit
340 N. Main Street
Suite 204
Plymouth, Michigan 4817
(313) 459-8730

Washington, D.C.
1730 North Lynn Street
Suite 400
Arlington, Virginia 22209
(703) 522-2118

ll Center II
Central Expressway
Texas 75206
91-8565

rk
Plaza West-I
Brook, New Jersey 07662
68-9471

ATES
a
Australia
Centre, 7-9 Merriwa St.,
110,
N.S.W. 2072
-8199
A 24434

ema SRL
ilano
Via Soperga 36
Italy
Milan 284-2850

7-7-26 Nishi-Shinjuku
Tokyo
Japan 160
(03) 371-3082

INPUT Planning Services for Management

IMPROVING MANAGEMENT EFFECTIVENESS THROUGH DECISION SUPPORT SYSTEMS

MANAGEMENT ISSUE REPORT

SEPTEMBER 1981

IMPROVING MANAGEMENT EFFECTIVENESS THROUGH DECISION SUPPORT SYSTEMS

CONTENTS

IMPROVING MANAGEMENT EFFECTIVENESS THROUGH DECISION SUPPORT SYSTEMS

EXHIBITS

1 INTRODUCTION

I INTRODUCTION

- The term, "Decision Support System" (DSS), is increasingly being talked and written about. Is DSS just the newest buzzword, or is it something more real that should be of interest and concern to Information System (IS) management?

- As this report shows, the concept and definition of a decision support system are still in a state of flux and are to a certain extent not consistent from one authority to the next.

  - However, there is a definite, real core of meaning that may well have a significant impact on data processing as well as on its practitioners.

- The information and analysis in this report come from the following sources:

  - Custom studies which INPUT conducted within the last year in several of the areas encompassing decision support systems.

  - Interviews with a selection of firms with active DSS programs, whose experiences have been incorporated as appropriate.

  - A review of the theoretical and academic literature on the subject.

  - Interviews with several of the leading vendors of DSS software and services.

  - Dialogs with specialist DSS consultants.

II WHAT IS A DECISION SUPPORT SYSTEM?

II WHAT IS A DECISION SUPPORT SYSTEM?

A. ACADEMIC DEFINITIONS

- Definitions are important to the extent that they promote understanding. In the case of DSS the very profusion of definitions that has come from commentators and academics has led to contradiction among the definers and puzzlement among the spectators. Examples of DSS definitions include:

  - A computer system to support unstructured or semi-structured decision-making.

  - A computer system with the following characteristics:

    . Extensible.

    . Able to support ad hoc data analysis.

    . Future-oriented.

    . Available for irregular and unplanned use.

  - A computer system made up of the following components:

    . A language system.

  - . A "repository of problem domain knowledge" (i.e., data).
  - . A processing system.
- A computer system developed by learning and adaptation.
- Computer-based systems that:
  - . Assist managers in their decision processes in semistructured tasks.
  - . Support, rather than replace, management judgment.
  - . Improve the effectiveness of decision-making rather than its efficiency.

- These definitional issues are well summarized, at much greater length in a forthcoming book by Ginzberg and Stohr, excerpts of which were presented at the NYU Symposium on Decision Support Systems, May 1981.

- Perhaps the most helpful academic said:
  - "There is at present little consensus about what qualifies a system as a decision support system."
  - Naturally, the person went on to provide yet another definition.

- In a way it is like the nine blind men and the elephant: they are all on to something, but what it is exactly is not clear.
  - INPUT's research and analysis leads it to conclude that there is a very big DSS elephant.
  - It is an elephant, moreover, that might crush the unwary IS director.

B. CASE STUDY EXAMPLES

- Before returning to more abstract definitions it will be useful to give an idea of the range and concreteness of what is termed a DSS today by including four representative case studies out of a number of self-described decision support systems which INPUT identified during its research.

1. ELECTRONICS MANUFACTURER: SALARY ADMINISTRATION

- This firm required an objective means of distributing its pool of merit increases. Previously, this function was performed manually by the personnel manager and engineering managers and required long hours of allocation and evaluation.

  - The manual method was so slow that only one iteration of the process was usually possible, with little or no refining. Often the initial merit increase budget would be revised and the whole lengthy process would have to be done again.

  - The merit package DSS allowed managers to perform as many iterations of their analysis as they felt necessary. It also allowed them to respond quickly and with little effort to budget modifications.

- This DSS was developed at the request of the personnel manager who approached the head of Information Systems Planning, who then enlisted the head of Operations Research to do the job. An Operations Research (OR) analyst with some free time was assigned to the task.

  - The OR analyst used APL to build the system in two to three months, from first discussion of the system to initial results.

  - It was possible to build this application quickly because it required only part of one person's time, and the OR head was willing to commit him and, equally important, the firm's normal formal review and approval process was not required.

  - The OR analyst was not a professional DP person, but had picked up enough APL to do the whole job himself. Writing the system did not require a great deal of detailed logic. The matrix capabilities of APL were quite helpful.

- The manager-users were active participants in the design and development of the system. The OR analyst performed the actual work of writing the system.

- The final system that was developed also:

  - Used a small amount of data extracted from the corporate personnel system. No external data were used.

  - Ran on an internal timesharing system.

  - Cost between $15,000 and $20,000 to develop, and ran with minimal operational costs.

- The firm reported no major problems in developing the DSS. The only problem was with finding the time for the user group and the analyst to get together to work on the project.

- The major benefits of the DSS to the firm are:

  - It improves management morale by keeping managers from having to work day and night during the merit review period.

. It frees managers to deal with other crucial matters (an opportunity cost situation).

. It allows managers to more fairly allocate merit increases because they are no longer restricted by a time factor.

. It improves the managers' performance.

2. MAJOR OIL COMPANY: FIVE-YEAR OPERATING PLAN

- This is a system that projects a five-year operating plan at the subsidiary level and consolidates subsidiary data into a total corporate plan. It gives each user the capability of doing "what if?" analysis at the subsidiary level and seeing the impact on the corporate bottom line.

- This DSS was built specifically for the Corporate Planning manager so that he could respond to queries from senior management.

- The Corporate Planning manager works interactively at the terminal himself to test alternatives and sensitivities.

- The interface to the system is simple and requires no technical ability.

- The system was built by the DSS department which is part of the Information Systems department and is headed by a DSS manager.

. Presently the DSS department is thought of as a facilitator organization and, as such, has no technical staff.

. The DSS department must rely entirely on outside consultants to build its decision support systems. This has led to problems in maintaining existing systems and has caused the DSS manager to request internal staff.

  - All resources for DSS at this firm come out of the functional managers' budgets. Only nominal approval is required from internal DP.

  - The DSS manager, who came to the job a year earlier, picked one application as a showcase and put all his efforts into making that a success.

  - It has been a success and, consequently, he expects little trouble in selling the DSS concept in other areas.

- The planning manager offered strong input in the initial development of the DSS and also made numerous suggestions concerning the prototypes of the system generated about every two months.

- The major tools used in building this system were:

  - RAMIS.

  - EMPIRE.

  - TELEGRAF.

  - Various statistical packages.

- The most important features of these tools, according to the firm, were their:

  - User friendliness.

  - Flexibility and speed in building and evolving new systems.

  - Ability to integrate database, modeling, graphics, and statistical capabilities.

- This DSS was developed over a six-month period from start to usable version.

  . Development cost was $30,000. Operational cost of the system is minimal, at about $2,500 for the three months during which the plan was created.

- The major benefits of this DSS are:

  . Divisional management is made more effective by allowing it to hypothesize modifications of divisional activity and see the effect on the corporate bottom line.

  . More alternatives can be explored.

  . More timely decisions, with better information, can be made.

3. AIRLINE: STRATEGIC MARKETING ANALYSIS AND RESEARCH

- This is a large DSS which uses an extensive data base of company and industry marketing, pricing, traffic, scheduling, and aircraft data.

- The ultimate end users of this system are the senior officers of the firm who request and receive ad hoc reports about particular decisions.

  - The direct end users of the DSS are first- and second-level managers, analysts and, in some cases, clerical personnel.

  - The data base for this DSS was developed through strong user input as it was created by a joint user/DP staff study.

  - Managers across the organization in a task force developed the data base, down to specification of individual data structures.

- The data base was pulled together by the Computer Services department. The user programs are developed by the end users with support from Computer Services.

- The software tools used in the system are:

  - FOCUS: Useful because it can interface well with data structures that are not well designed. This feature is needed frequently when newly developed or external data are added to the system.

  - SAS: A good advance statistical tool.

  - SIMPLAN: Used for spread sheet analysis.

  - PROJECT II: For large project scheduling.

  - MARK IV: Used to support existing batch oriented processes.

- The most important features of these tools, according to the company, are:

  - The nonprocedural nature of the user interfaces.

  - The ability to access many different data structures in many different operating environments.

  - Product efficiency and vendor service and support.

- This is a large system:

  - It cost $1 million to develop and generates operational costs of about $2 million per year.

- The concept and design have allowed the system to evolve over time to meet the needs of different users. (There are now about 600 users of the system.)

- The firm feels there have been major benefits from the DSS:
  - Better analysis.
  - More alternative views of decisions.
  - More timely support of users.
  - Broader dissemination of data (both local and global).

4. CONSUMER PRODUCTS COMPANY: MARKETING PROMOTION

- This firm is heavily involved in consumer marketing and spends millions of dollars on marketing promotion and advertising. The company wanted to analyze how it was spending its promotional money to determine if it could be spent more effectively.
  - Questions arose such as to whether it was possible to spend less on promotion and still sell as much, or sell less but increase net income.
  - The overriding question was "How much promotion should we do and what is the timing?"

- The divisional president of this firm is the ultimate decision-maker. The direct users of the DSS are the marketing managers, brand managers, and the promotional department.
  - This DSS was implemented by an outside consultant, with the system users playing an important role in its conceptual design.

- Within the firm, the DP department is responible for integrity of internal data and, as such, was consulted for approval for releasing the data to users.

  - DP also monitors computer usage by end users and consults with management when usage appears excessive.

- The consultant for this project reported to INPUT that the addition of new data to the system with NOMAD proved to be quite easy. In-house data were constantly being stripped off existing files and integrated into the DSS.

  - Some users of the system were capable of doing this themselves. In other cases the consultant did it.

    . Many users also added some of their own data at the terminal.

  - This DSS cost less than $20,000 to develop, including consulting fees. The consultant on this project reports that users spend between $50 and $75 per hour to run the system.

    . The system is run on an outside timesharing service and fees for this service, as well as for the consultant, are funded through the user departments' outside services budget.

- The DSS is considered a major success within the firm since promotional savings of $6 to $8 million a year have been generated without any loss in sales. This came about because:

  - There was now the ability to look at many alternative promotional possibilities.

  - The firm can now also look into many other areas that could affect promotional strategy; e.g., the effect that product price changes might have on promotional policy.

- In addition, there was a qualitative benefit in that managers and analysts now have to do far less clerical "grunge" work and thus can do more analysis. With lower frustration levels, these people do higher quality work, according to the company.

C. A PRAGMATIC DEFINITION

- These are certainly examples of interesting, useful computer systems. But what have they in common? Can we even make some preliminary judgment as to what is not a DSS?

- Those who would set up hard and fast abstract rules as to what a DSS is (or is not) will do so at their own risk.

  - At a recent DSS conference the keynote, overview speaker gave inventory systems as an example of a class of computer systems that would not qualify as DSS (because they were virtually automatic in operation, requiring negligible human intervention and judgment).

    . However, two of the invited speakers proceeded to give as case studies examples of DSS that were inventory systems.

    . All speakers had valid points to make!

- The chief problem with the academic definitions cited earlier is not that they are untrue (since they are usually valid in their own ways) but that they are not oriented to the world of the data processing practitioner.

1. INPUT'S DEFINITION

- In this section INPUT will define the complex of characteristics of a DSS in practical and pragmatic terms. The main identifying features are:

- Overall importance.

- User characteristics.

- Flexibility.

- Timeliness.

- Uniqueness.

- Kind of data needed.

a. High Importance

- The chief thing to keep in mind is that these are important functions - so important that they will be carried out somehow.

  - Acquisitions or five-year plans will go ahead, with or without a DSS or a computer system, if someone high enough wants them. The alternative may be using either the back of an envelope or many clerk-years.

  - It is the perceived importance of the work to be accomplished and the awareness of the inadequacies of manual alternatives which impel the creation of many decision support systems.

b. Senior User Initiated

- The importance is to end user departments - they initiate action on DSS development.

  - It is not any user, but generally a senior executive.

    . The head of payroll, but rarely the finance vice president, cares about the payroll system.

. The finance vice president will care about "his" DSS.

- Not only does senior management initiate DSS development, it is also the ultimate user of such systems.

. This is usually not so in the keyboard or coding sense, but is definitely so when it comes to examining outputs and taking part in reformulating the approach.

c. Flexible Development

- This is directly related to another key characteristic: initial expectations and project requests may assume that a particular system will fulfill expectations.

- It is in fact very rare that this will happen: initial results will usually be the first in quite a long line of intermediate results. This is what the academics mean by "learning and adaptation."

- Very often an explicit model is involved, with the key relationships tinkered with even before different assumptions are fed through the model.

. Models are often implicit; a conventional appearing DP system may be constructed but using a "bread board" approach, to see how it works, then the system may be modified.

- This key characteristic, of system iteration and evolution, is in fact a good approach to follow in constructing "conventional" systems.

. It is not done because of perceived time and resource constraints as well as unfamiliarity with newer software tools (e.g., INQUIRE, FOCUS).

- In addition, there is often a blind faith that "rigorous analysis" will identify all needs and the best way of meeting these needs, once and for all. This is sheer prejudice and is almost always proved wrong by the events that follow.

  - MIS departments should follow with great attention the outcome of these approaches and be prepared to build on successful experiments.

- There is the further possibility, always present, that a DSS once "set," will change from top to bottom in terms of inputs, logic, data, or form of presentation.

  - Decision support systems are very sensitive to changes in the external environment, since they are one way that organizations try to better adapt to and control the outside world.

  - Changes in markets, competitors' products, internal costs, laws, tax treatments, or company policies and assumptions, can all have pervasive effects on a DSS.

d. Fast Development

- While the requesting department wants lots of opportunity to play around with the system it also wants the whole system ready quickly.

  - Very often the user is working against a timetable that has been imposed on him or, as in the case of an acquisition analysis, yesterday is too late.

  - Bright ideas and targets of opportunity cannot wait for feasibility studies, programmer availability, or COBOL debugging.

    . This is why timesharing services estimate that one-third of their business comes from DSS-type work. They are ready. It helps that in these situations money is literally no object.

e. High Degree Of Uniqueness

- DSS systems are virtually always unique.

  - They are highly dependent on the coming together of:

    . A particular person.

    . The organization's needs at a point in time.

    . The particular factors in the external environment currently deemed critical (e.g., interest rates are now a much more critical factor than they were, say, ten years ago).

    . Data availability.

    . The resourcefulness of the DSS builder.

  - While the salient features and even most details of a payroll system are identical from firm to firm, this is not so of decision support systems, even those with a superficial resemblance or similar names (e.g., acquisition analysis).

f. Data Dependency

- One of the reasons for this uniqueness is the relationship of a DSS to data.

  - A DSS is very data sensitive: it usually feeds on live data. Without precise data a DSS merely states the possibility of interesting relationships occurring, but users cannot know how these relationships affect them.

  - A DSS rarely requires new data to be generated from company sources. In fact, data are usually extracted or summarized before being used.

  - One service that a DSS may perform is to make clear just how unclean so much of a company's operational data are. The data simply cannot stand up to analysis and, therefore, cannot support analysis.

  - Sometimes a successful DSS will lead to a reevaluation of data capturing, processing, and organization so that, among other things, better company decisions can be made.

- Decision support systems are increasingly using data from a wide variety of external sources. This is driven by two intertwined forces.

  - Decision and models increasingly have to take into account facts about the outside world.

  - These data are increasingly available in machine readable form, often integrated by the same timesharing firm which supplies and supports the DSS. Exhibit II-1 gives an indication of the depth and breadth of these public data bases.

2. CONTRASTS BETWEEN DSS AND TRADITIONAL SYSTEMS

- It is useful to summarize and contrast the differences between a decision support system and a traditional system, as shown in Exhibit II-2.

  - Traditional systems go through a long development process and are used by fairly low-level people. They are often vital to a company's operations, but this is usually recognized only when they do not work.

  - Decision support systems are frequently the focus of bursts of high-level activity (sometimes, alas, misinformed and misdirected). It is reminiscent of data processing of the 1960s.

EXHIBIT II-1

EXAMPLES OF FINANCIAL INFORMATION DATA BASES (PARTIAL LISTING)

- Cates Lyons and Company maintains a historical data base of over 800 key financial data items on 250 major bank holding companies.
- Robinson-Humphrey Company maintains a data base of key financial items on 145 top bank holding companies. The data base is offered together with comparative analysis software.
- SBC maintains a financial institutions data base of financial information containing:
  - FDIC data on over 14,000 commercial banks.
  - FHLB data on over 4,500 savings and loans.
  - NCUA data on over 16,500 credit unions.
- Payment Systems, Inc., offers a data base through IDC containing statistics on major aspects of financial transaction systems, including ACH, ATM, credit cards, NOW and share draft accounts, and telephone bill paying systems. The data base also includes key money market indices and market attitudes data on both electronic and paper payment systems.
- Blyth Eastman Dillon & Company maintains a financial data base that contains daily price and yield information on over 800 bonds and other money market instruments including U.S. Treasury notes.

SOURCE: ADAPTED FROM INPUT's REPORT, MARKET OPPORTUNITIES FOR DATA BASE SERVICES

EXHIBIT II-2

SUMMARY OF DIFFERENCES BETWEEN DSS AND NON-DSS

FACTOR	DSS	NON-DSS
Senior personnel initiate?	Yes	Not usually
Senior personnel use?	Yes	No
Timeframe	Short	Medium to long
Changes in software design/ coding assumed?	Yes	No
System reused regularly?	Sometimes	Usually
Model-oriented?	Yes	No
Off-the-shelf packages usable?	Rarely	Usually
New internal data elements created?	Rarely	Often
External data required?	Often	Rarely

III WHAT IS NEEDED TO MAKE A DECISION SUPPORT SYSTEM WORK

III WHAT IS NEEDED TO MAKE A DECISION SUPPORT SYSTEM WORK

- In the previous chapter INPUT showed the key characteristics that distinguish a decision support system from other kinds of data processing systems. Omitted from that discussion were the important areas that actually enable a DSS to work, including:

  - The components of a decision support system.

  - DSS software.

  - Hardware.

  - Organization issues.

- These are all key factors for making a DSS actually work. Obviously, they have much in common with other types of data processing systems. However, as will be seen, there are many uniquely DSS issues involved.

A. DSS COMPONENTS

- In principle, a decision support system is the same as any other computer-based system: there is input-processing-output.

- Even DSS specialists describe the system building steps as very similar to the steps in a traditional computer system, as shown in Exhibit III-1.

- However, on closer inspection a decision support system contains components not usually given prominence in traditional systems, with different types of relationships existing between them (at least conceptually).

- In Exhibit III-2, the major components are broken out into:

  - "Operators," which tell the system what to do.

  - Functions.

  - Data.

I. OPERATORS

- What INPUT has termed "operators" are the heart of the DSS.

  - Processing logic is typically supplied by the user for each new DSS.

    . This may be within the context of a software modeling package, with or without menus or "fill-in-the-blanks" which make use easier for many people.

  - A query language/report writer (which may be one or more software tools) is a critical element in a DSS and an important part of its user-friendliness (or lack thereof).

  - Much effort is devoted to making this part of some DSS software tools as easy to use as possible, since the target user is assumed to have sketchy DP background and will, in any event, usually not be working full time with the DSS software.

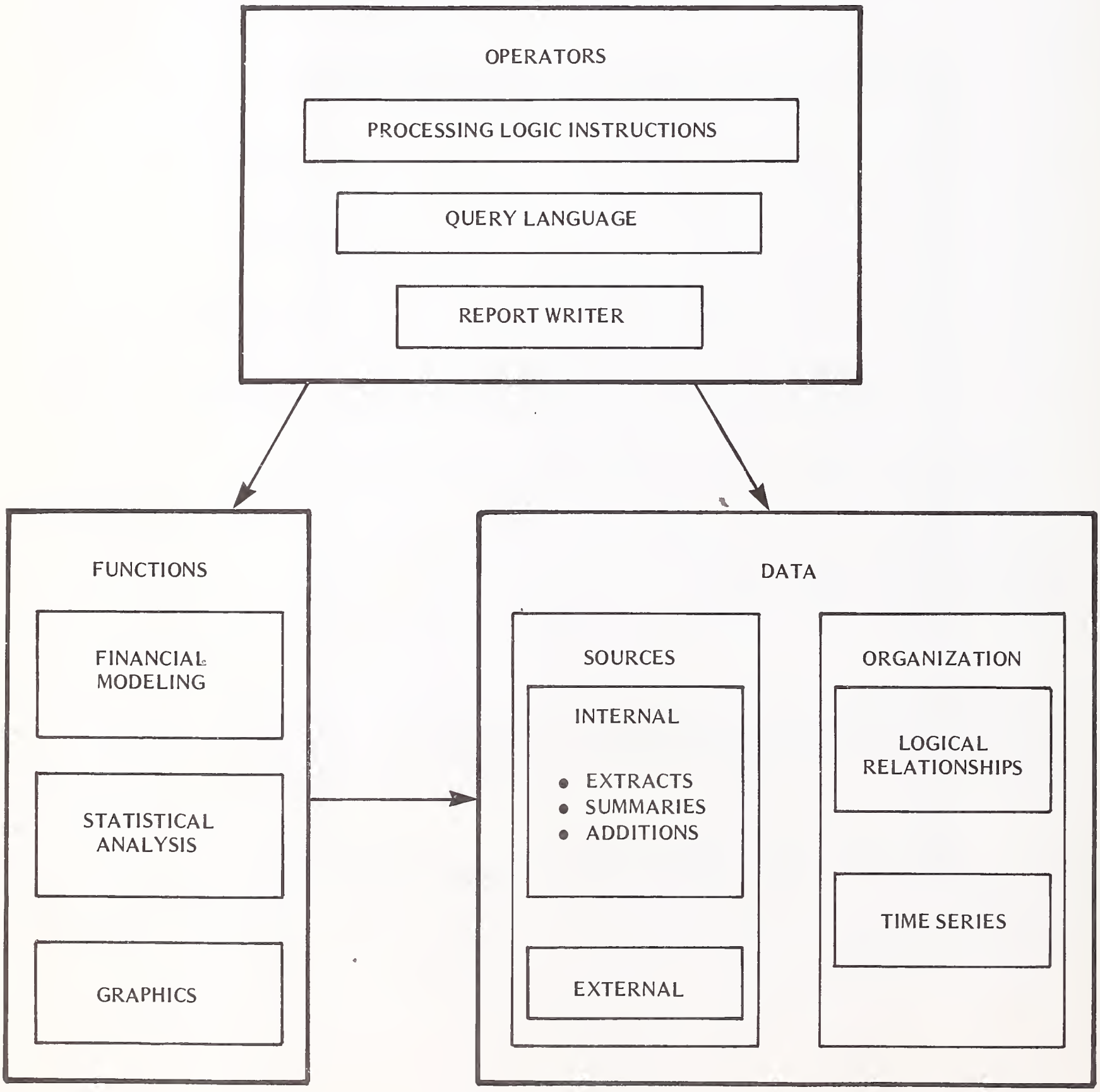
EXHIBIT III-1

FIFTEEN STEPS TO ACHIEVE AN INTEGRATED DSS

1. Establish management's needs
2. Identify system tasks
3. Prioritize tasks
4. Identify system resources
5. Write functional specification
6. Define data element dictionary
7. Write external specification
8. Write internal specification
9. Develop test data
10. Code system
11. Write user's guide
12. Catalog system modules
13. Write system maintenance manual
14. Test system
15. Write application guide

EXHIBIT III-2

DECISION SUPPORT SYSTEM COMPONENTS



2. FUNCTIONS

- Some engineering and scientific systems have similar libraries of functions to draw on, but such accessible functions have until recently been fairly uncommon in business-oriented systems. Exhibit III-3 gives a sense of the range of functions available.

  - Not all functions will be required in all applications. In fact most departments using a DSS will tend to use a limited subset of these functions.

    . However, as new analytic tasks are taken up, new functions will be necessary.

    . Staff transfers, seminars, etc. can also affect which kinds of functions are used, even on existing applications.

3. DATA

- Data issues can become quite complex for two reasons.

  - There is a multiplicity of data sources.

  - Data organization is more complex than most operations-oriented data bases.

- Exhibit III-4 shows the possibilities for different types of data to be used for a decision support system. Many large companies build just this kind of corporate DSS data base.

  - Simply keeping all the updates in synch is a problem.

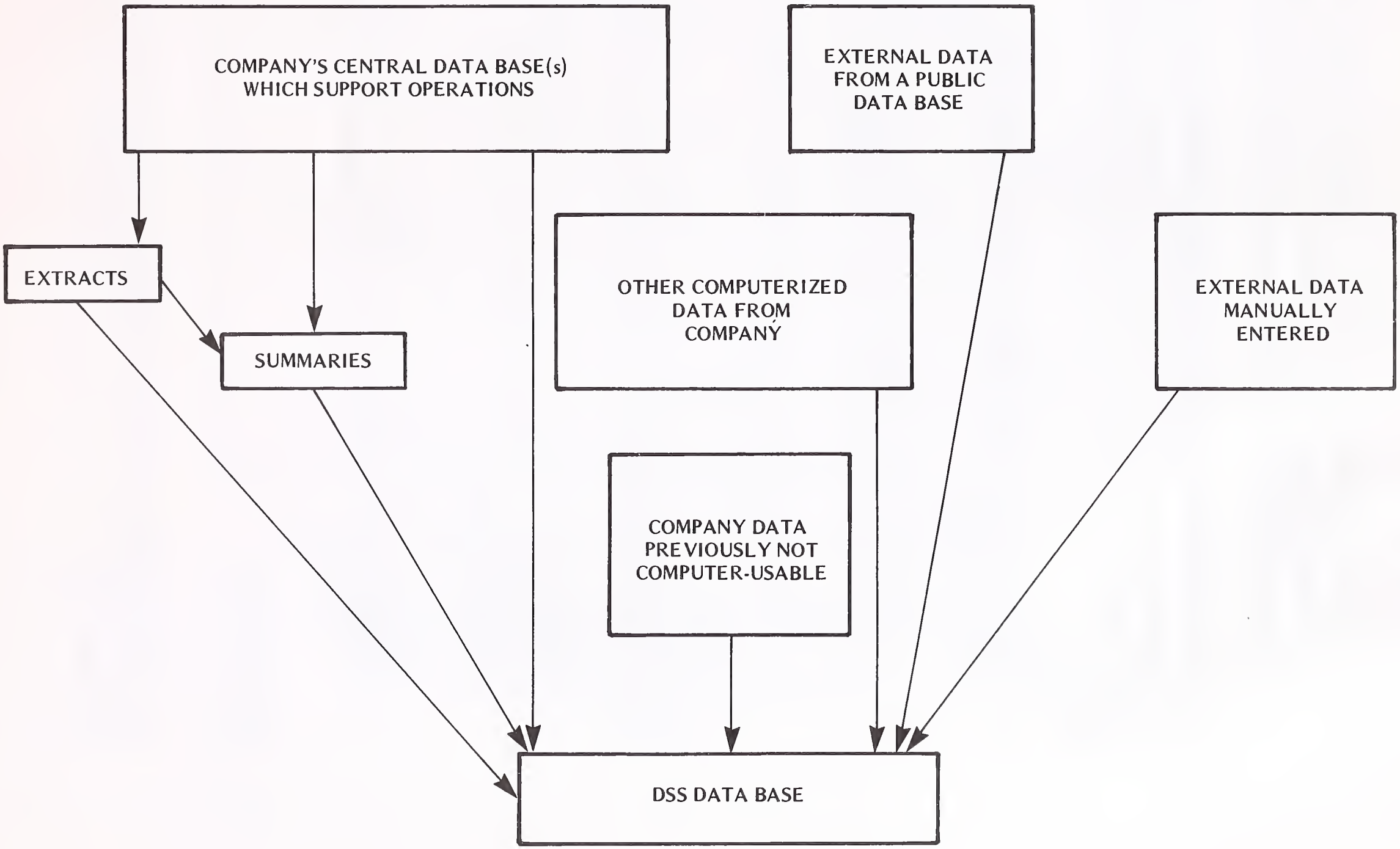
EXHIBIT III-3

EXAMPLES OF DSS FUNCTIONAL CAPABILITIES

- Amortization
- Annualization
- Backward iteration
- Built-in distribution functions
- Compound interest
- Curve fitting
- Depreciation
- Discounted cash flow
- Equation reordering
- Exponential smoothing
- Financial ratio analysis
- Forward referencing
- Impact analysis
- Lease/purchase
- Linear regression
- Monte Carlo simulation
- Multidimensional variables
- Multilevel consolidation
- Multiple regression
- Net present value
- Pro forma capabilities
- Risk analysis
- ROI
- Significance testing
- Simultaneous equations
- Spreading
- Time-series forecasting

EXHIBIT III-4

SOURCES FOR BUILDING THE DSS DATA BASE



- A more insidious problem is that involved in keeping the logical relationships correct between elements in the DSS data base if data elements change their meaning somewhat from their original source.

- DSS data bases are different from most operations-oriented data bases in that time series are very important. This is understandable since many decision support systems are engaged in trying to foresee future events based upon past data.

  - Many commercial time series oriented data bases have grown up to meet this need.

  - It is often difficult to construct an initial time series of internal data, since such time series data are rarely used for operational purposes.

    . Changing data structures, data definition, and data quality make this a nontrivial task.

- Many DSS constructors would wish, ideally, to have what might be termed a "three-dimensional" data base, the concept of which is shown in Exhibit III-5.

  - Perhaps this will be the next stage in relational data bases (presumably requiring very large processing and storage overheads). This will be some time away.

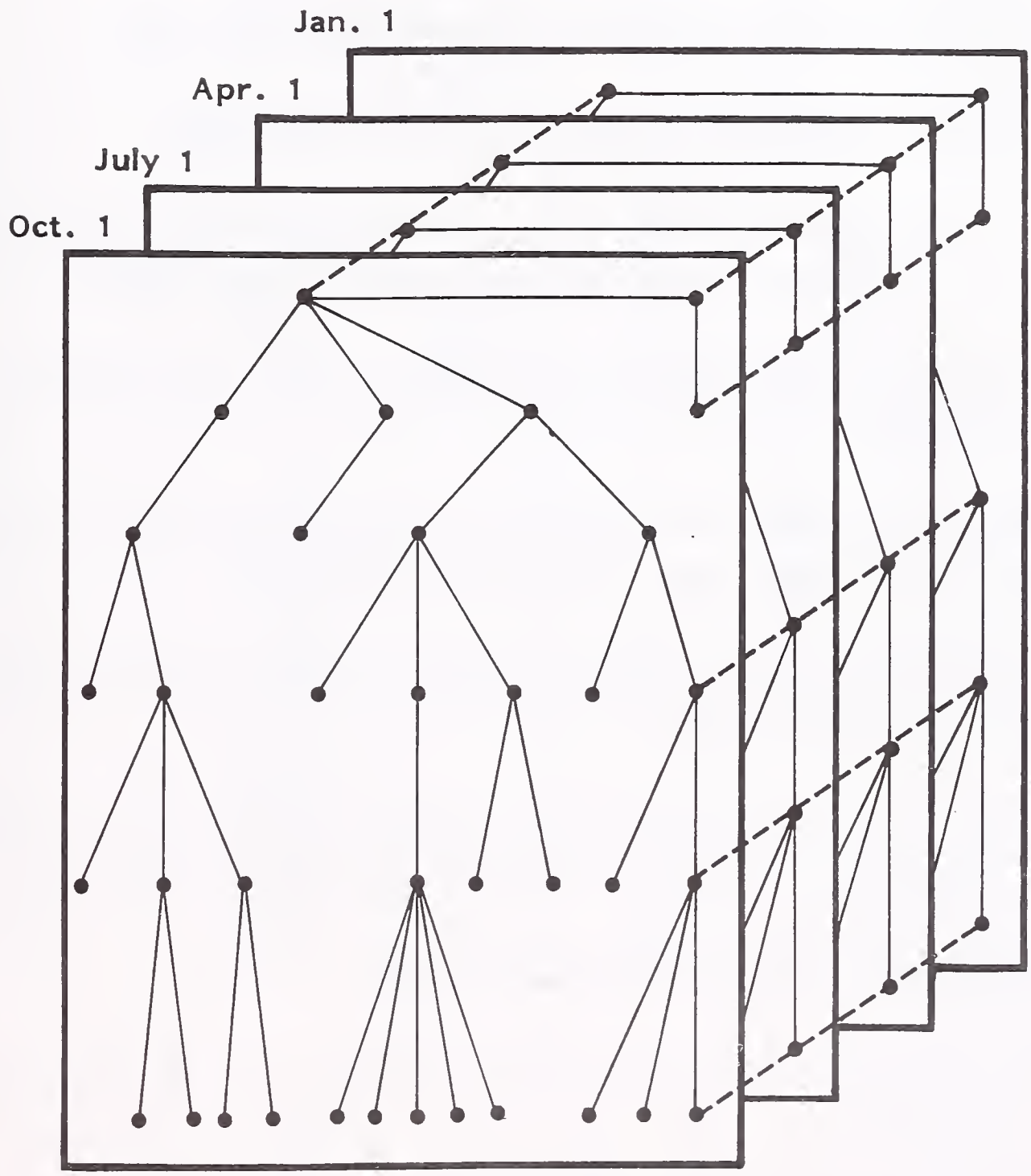
  - Right now, there is usually a choice that must be made between handling either logical relationships or time series well (i.e., easily).

B. DSS SOFTWARE

- Software encompasses both the "operator" and "function" components described earlier.

EXHIBIT III-5

THE THREE-DIMENSIONAL DSS DATA BASE:
LOGIC AND TIME



——————— Logical relationships at a single point in time

- - - - - - - Time series

I. SOFTWARE TYPES

- There are approximately 100 languages/packages used for DSS purposes. They fall into five general categories:
  - Traditional programming languages (e.g., FORTRAN, BASIC).
  - Newer, more specialized programming languages (e.g., APL).
  - "Fourth Generation" languages (e.g., FOCUS, INQUIRE).
  - "Home made" DSS packages (e.g., FORTRAN, a statistical package, and an already existing operations-oriented DBMS, such as TOTAL).
  - Vendor-supplied DSS packages (ranging in price from VISICALC to EXPRESS).
- Generally, the performance characteristics of the packages within each group cluster around the same value, as shown in Exhibit III-6.
  - In general there is a tradeoff of price and hardware efficiency against ease of use and features.
  - The low price of VISICALC and its imitators introduces some anomalies into the matrix. This issue will be discussed at greater length below.
- The multivendor, home-grown approach is common.
  - It often represents a compromise that manages to get the worst of both worlds:
    - Inefficient use of hardware resources.
    - Difficulty in obtaining support.

EXHIBIT III-6

CHARACTERISTICS OF DSS SOFTWARE APPROACHES

TYPE OF APPROACHES:	TRADITIONAL PROGRAMMING LANGUAGE	NEWER PROGRAMMING LANGUAGE	"FOURTH GENERATION" LANGUAGE	"HOME-MADE" DSS PACKAGE	VENDOR SUPPLIED DSS PACKAGE
Example of approach:	FORTRAN, BASIC	APL	FOCUS, INQUIRE	DBMS + language + statistical package	VISICALC, EXPRESS
Characteristics:					
Price	Low	Low	Medium-High	Medium-High	Low-High
Hardware resource consumption	Low	Low	High	Medium-High	Medium-High
Ease of nonprogrammer use	Low	Low/Medium	Medium	Low-Medium	Medium-High
Features-range	Low	Low	Low/Medium	Medium-High	Medium-High
Features-integration	Low	Low	High	Low-Medium	High
Coding speed	Low	High	High	Low-Medium	Medium-High
Modifications ease	Low	Low-Medium	Medium	Low-Medium	High
Modifications - turnaround speed	Low	Low-Medium	Medium	Low-Medium	High

. Limited features.

. Relatively unfriendly.

- However, it represents a perceived low entry price approach since often all that has to be acquired is a statistical package (e.g., SAS).

- The "fourth generation language" approach makes a lot of sense where a "mini-system" (or perhaps even a large system) has to be put into place (to collect and store data, for example) before the DSS per se can begin to function.

2. VENDOR PACKAGES

- Increasingly, DSS software means one of the integrated packages that have been developed over the past ten years. This is because many of these packages fill an important need at a reasonable cost.

  - To a certain extent, however, the very existence and active marketing of the packages have helped to create a demand for them (and, possibly, for the DSS approach).

    . "Marketing" does not mean just advertising and sales calls; as a top executive at one of the leading firms told INPUT: "We can create primary demand through professional activities; advertising just builds brand preference."

- If a company intends to seriously engage in modeling and other DSS activities, INPUT does not recommend a do-it-yourself approach, either by relying on a programming language, or by trying to construct its own model (in, say, FORTRAN).

- In the early and mid-1970s many companies tried the in-house approach. By now, however, even some of the largest companies are abandoning it.

- They are writing off often formidable past investments because the in-house approach is not friendly enough and cannot be maintained.

- The up-front and ongoing investment in building a DSS package system should not be minimized.

  . One leading firm estimates that the current product took 75 man-years to develop.

  . Another believes that not a single line of code written five years ago remains in the present version of the package.

- A single company, no matter how large, will generally spend less to provide itself with as many copies as are necessary of a proprietary product.

3. SELECTING A VENDOR PACKAGE

- All prospective vendor package buyers want to know: "What's the best package?"

  - As in the case of jogging shoes the only truthful answer is, "it depends."

- There are two basic reasons for this answer:

  - Package offerings are constantly changing, as products enter and leave the market and, more importantly, as products are modified and enhanced.

  - Even more than most packages, each DSS package tends to have its special strengths (and, perhaps, weaknesses).

    . These should be closely matched against what the purchaser sees to be its key needs.

- Two key objective discriminants are price and product age.

  - Very expensive ($100,000 plus) packages are much more flexible and often have more features than, say, most $3,000-10,000 packages.

    . VISICALC, at $100-200, is in a class by itself.

  - Recently released products tend to have learned from earlier products and offer more.

    . However, they may have start-up problems, and they may not stay the distance.

  - However, in the broad middle range of products (in terms of price and age) the tradeoffs are very complex and each user will have its own special needs to be matched against products.

- An observation made by several vendors is that there are too many vendors already, given the current size of the marketplace.

  - This means that commitment to the market as well as sales growth and profitability should be kept in mind by any purchasers.

- Based upon INPUT's observations, there are several key areas to focus on in defining the company's needs for a DSS software package.

  - What features are <u>really</u> needed?

    . Too often, technicians want "one of everything"; this will add to the cost and complexity.

    . A few features (e.g., consolidations) may be so important that the search will really revolve around them.

- Will there usually be a few or many sources of data?

  . If much external data are needed, then it would make sense to deal with a timesharing firm that could supply both the data and the software package.

- Will logical or time series data be most important?

- How self-sufficient will the organization (both users and the MIS department) be?

  . This will determine the importance of local support and documentation.

- Is a large volume of data anticipated?

- Benchmarking realistic jobs is appropriate if major activities can be defined in advance.

- Heart to heart talks with current users can substitute for benchmarks and may be more revealing in many cases, given the complexities of setting up equivalent benchmarks.

C. HARDWARE

- There are six major options for providing hardware resources for a DSS, as shown in Exhibit III-7.

- Decision support systems with large data storage and manipulation requirements are generally limited to mainframe-based services (either RCS or in-house).

EXHIBIT III-7

HARDWARE ALTERNATIVES FOR DSS OPERATION

1. Remote computing service (RCS) timesharing vendor
2. Mini/microcomputer linked to (often supplied by) RCS vendor
3. In-house timesharing (supplied by large-scale mainframe)
4. Dedicated central site DSS computer (e.g., 4300) sharing mass storage
5. DSS department mini/microcomputers (e.g., DEC 20, VAX), standalone or distributed
6. Personal computer (e.g., Apple, TRS-80)

- Moderate-sized decision support systems have a much larger list of alternatives.

- Many IS departments want, almost instinctively, to bring work in-house and run it as central site timesharing (alternative 3).

  - This could cause problems where a DSS application required enormous, irregular system resources.

  - Some IS departments are planning to provide the large DSS users with their own machine, either at the central site or the user's site (alternatives 4, 5).

  - In the past the in-house hardware option has not competed well with the RCS option.

- Data requirements will often determine the hardware options.

  - Large in-house data requirements will make an in-house machine attractive.

  - The opposite is usually true where extensive external data bases must be regularly accessed.

D. ORGANIZATIONAL ISSUES

1. FUNCTIONAL ROLES

- Some popularizations of DSS make much of the example of a top executive sitting down at a terminal to interact with the system. Discounting a few notable exceptions, such as Ben Heineman of Northwest Industries, typically

top management is in only at the beginning of the hunt and at the kill, as far as technical interaction with the system is concerned.

- Middle management typically serves as a buffer between top management and the actual technician, although a number of technical interface options are possible. Exhibit III-8 shows the most common ways of having user personnel interact with the system. None is intrinsically better than the others. The key point is to keep top management involved with the substance of the work.

- Overall, there are six major decision support system activities:

  - Posing the problem or issue to be addressed.

  - Deciding on the approach to be taken.

  - Entering data and performing the actual coding or programming.

  - Working with the output (usually iteratively).

  - Recommending courses of action (either on the substance of the work or on technical issues).

  - Making the decision.

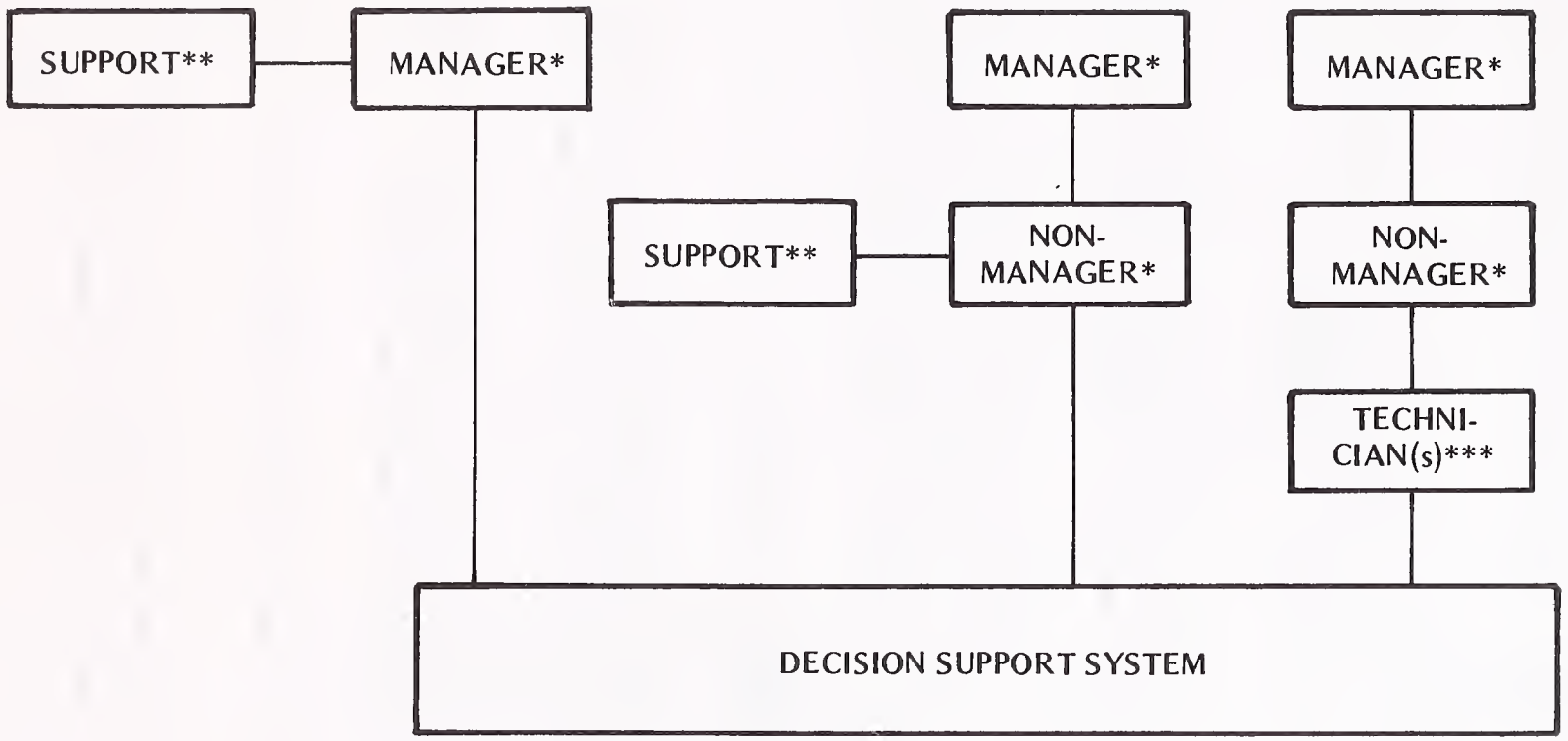
- Theoretically, people at different levels of the organization could be involved in any of the activities (and probably are in a few exceptional organizations). However, taking the four categories of personnel below, each tends to have activities they are (or at least should be) most involved with.

  - Top management.

  - Middle management.

EXHIBIT III-8

DSS INTERFACE OPTIONS



*USER DEPARTMENT
**VENDOR, CONSULTANT, AND/OR MIS DEPARTMENT
***CONSULTANT, MIS DEPARTMENT, AND/OR USER DEPARTMENT

- User analysts.

- Technicians.

- It is not, however, an on-off type of involvement.

  - While top management, for example, should be highly involved with posing problems, middle management should be involved also.

- For illustrative purposes it is useful to chart the level of involvement of the different personnel categories for the different activities. Exhibit III-9 shows the typical level of involvement.

  - It would be useful to graph, at least mentally, each particular organization involved with DSS activities and compare it to the norm.

    . There may be excellent reasons for variations; if there are, the reasons should be well understood and they should be functional and not accidental or exist for historic reasons.

2. ORGANIZATION OPTIONS

- The initiative and implementation of a decision support system are usually best placed in the actual department whose system it is.

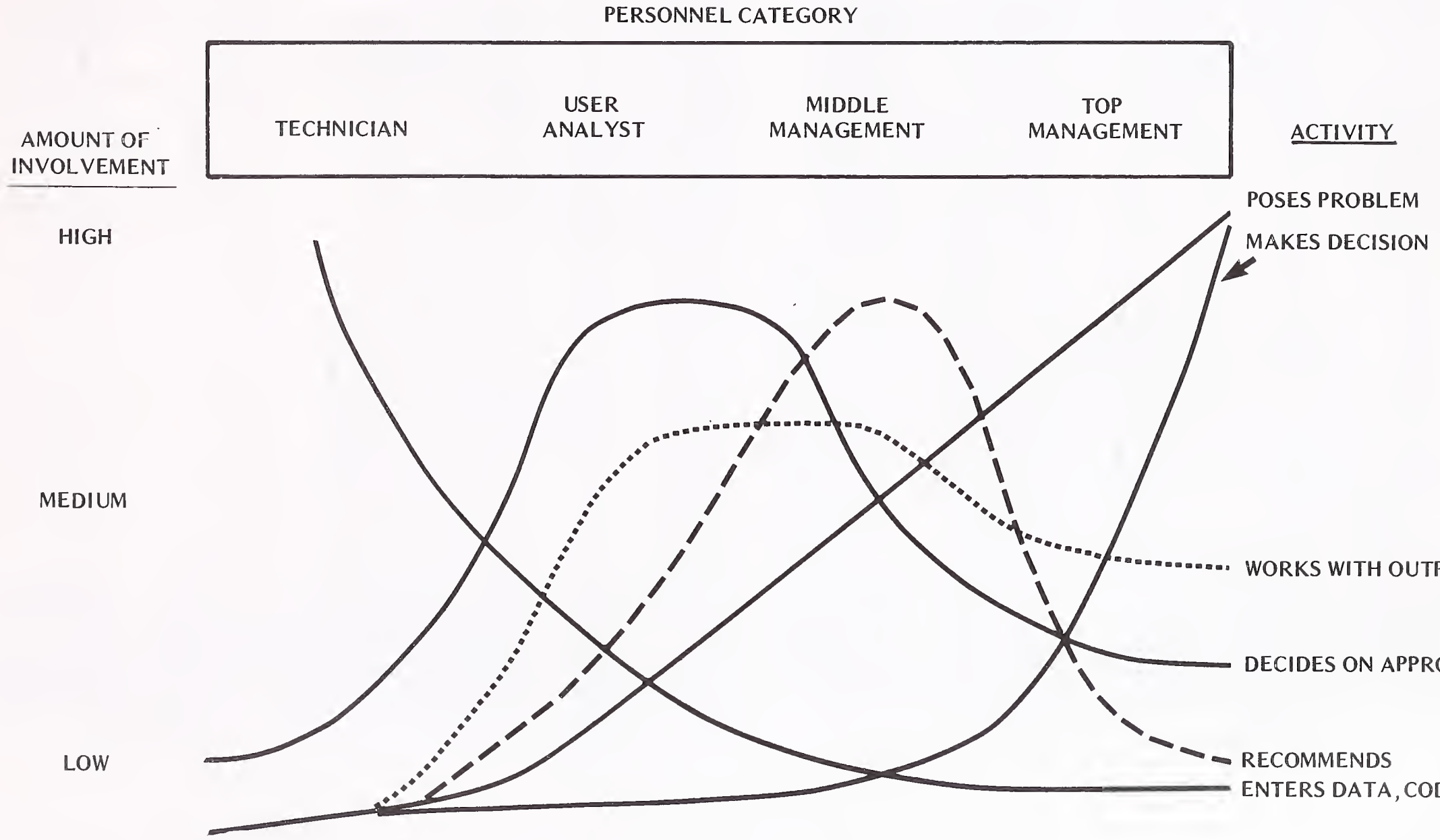
  - However, there is a whole range of support functions that may or may not be in the departmental unit.

- Currently, there is often no DSS support per se anywhere within a company.

  - The whole system and its maintenance may be in the hands of a consultant or similar external entity.

EXHIBIT III-9

AMOUNT OF INVOLVEMENT BY DIFFERENT LEVELS OF PERSONNEL IN DSS ACTIVITIES (SCHEMATIC)



- Often, major support functions are provided by the timesharing and/or software vendor.

- However, more mature and/or larger DSS tend to outgrow their total dependence on outsiders.

  - Outsiders are not always there when needed or, in the case of consultants, may sever their ties completely.

  - Timesharing and software vendor support become quite expensive for such relatively simple things as initial training, ongoing training, advice on functions to use, troubleshooting, etc.

  - In addition, over time, a unique repository of knowledge builds up (or should build up) around a particular DSS application, especially in the relationship of data and conclusions.

    - Organizing this body of knowledge to assist both technicians and decision-makers can greatly enhance the value of the DSS.

- Exhibits III-10 and III-11 show four of the major organizational alternatives.

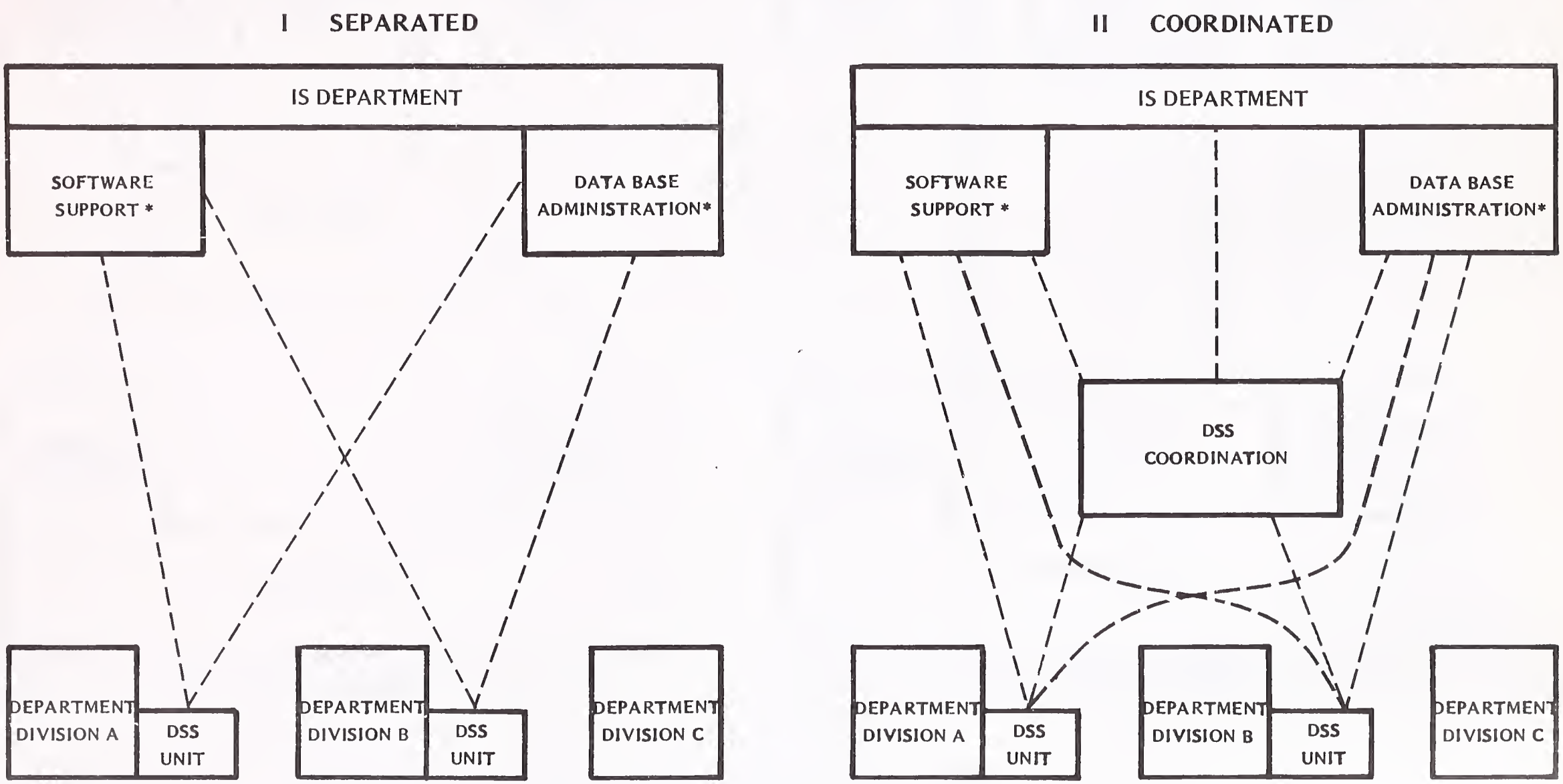
  - Alternative I of Exhibit III-10 has each division/department self-contained as far as support is concerned. This can be expensive and loses the benefits of cross-fertilization. (Of course, if the different units are very dissimilar they would be able to provide mutual support on only purely technical issues.)

  - Alternative II of Exhibit III-10 allows for a coordination unit (probably a single person), but otherwise the individual units are still self-contained.

  - In Alternative III (Exhibit III-11), there is an independent unit with its own staff of technicians and analysts that would be detailed to

EXHIBIT III-10

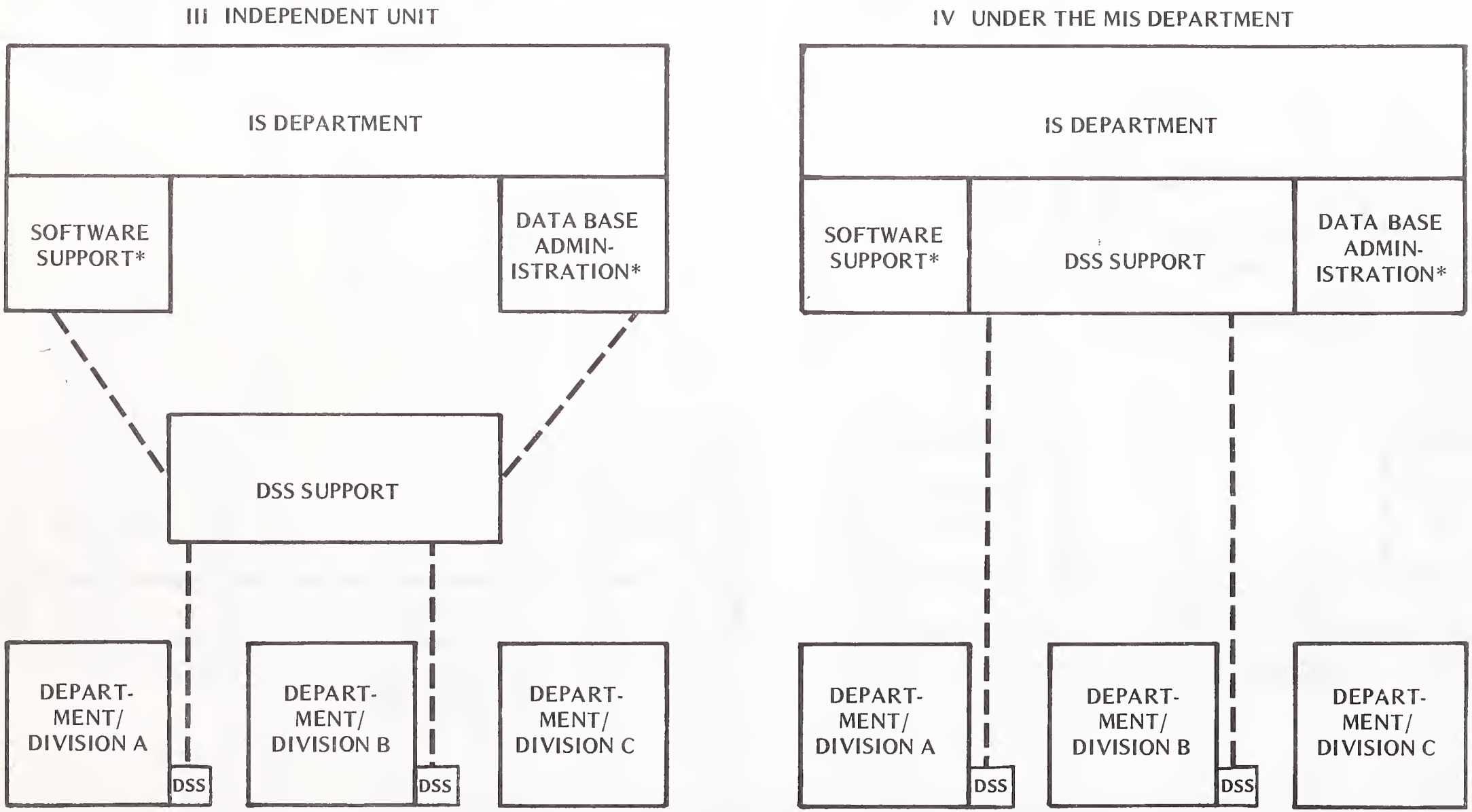
DECENTRALIZED ORGANIZATION OPTIONS FOR DSS SUPPORT



* ILLUSTRATIVE FUNCTIONS

EXHIBIT III-11

CENTRALIZED ORGANIZATION OPTIONS FOR DSS SUPPORT



*ILLUSTRATIVE FUNCTIONS

individual units. The staff would have to be carefully selected, especially the analysts who would need to have both a technical and business background as well as a flair for internal marketing. (A clear danger exists that these people would soon be hired by DSS vendors.)

- Alternative IV (Exhibit III-11) is similar to alternative III, except that the DSS support unit is part of the IS department. This would probably provide better technical coordination but has a number of dangers, whose criticality would vary, depending on the organizational situation.

  . The balance of knowledge and concern might tilt too heavily toward the technical and too far away from business analysis.

  . Some user departments might use IS-controlled assistance as a Trojan horse to take away control of "their" system and might back away from involvement, negating the benefits of a central support unit.

IV DECISION SUPPORT SYSTEM ISSUES

IV DECISION SUPPORT SYSTEM ISSUES

- There are a number of issues surrounding decision support systems which could have significant effects on the IS department:
  - The potential positive and negative impacts on IS.
  - The factors leading to DSS growth.
  - The factors that can cause DSS efforts to fail.
- Each of these points will be addressed in this chapter.

A. THE DSS IMPACT

1. POSITIVE FACTORS

- Many positive factors can flow directly out of DSS activity. These include:
  - New uses for data processing.
  - Real user involvement.
  - United forecasting, analysis, and operations data.

- Increased value-added data processing.

- Increased importance of data processing to the firm.

- The examples cited earlier are just some ways that data processing can be used in a DSS environment.

  - What was formerly guessed at or done on the back of an envelope can be done on a computer.

- Real user involvement is the hallmark of a successful DSS effort.

  - For a long time, many of the causes for ineffective data processing systems have been placed, correctly, at the door of faulty relationships with users.

- One of the less obvious effects of DSS activity will be the increased merging of data that are used for forecasting, analysis, and operations.

  - Right now, DSS data bases tend to be purpose-built and do not use the standard corporate operation data base, e.g.:

    . A DSS for personnel issues will usually not work directly with the corporate payroll-personnel data base for reasons of security, efficiency, and understandability; in addition, needed data are often not resident in the operations data base.

  - As DSS activity builds up, however, it will no longer be functional to maintain many separate data bases.

    . Similar pressures led to the construction and use of data base management software for operations-linked data in the last decade.

- In a DSS environment, a higher proportion of data processing activities will have a higher value associated with them. This is reasonable since management time (and related decisions) is worth more than clerical time.

  - Heretofore, much of data processing has consisted of mechanizing (and assisting) clerical activities.

- Finally, data processing will become even more central to a company's health and growth than it is now. This may take some time to be fully perceived.

  - Even in companies that have very large on-line networks supporting basically clerical activities, the importance of computer activities to the firm is often not widely appreciated.

2. NEGATIVE FACTORS

- Since there is no free lunch, it should be expected that there will be negative factors associated with decision support systems. These include:

  - Uncontrolled distributed data processing.

  - Less efficient production systems.

  - Data/data base confusion.

  - Unforeseen surges in hardware use.

  - Unsecured systems.

- Increased costs are not included as a negative factor, since the higher value added will make associated costs worthwhile.

- It will be even more difficult in a DSS environment to plan distributed systems than it is now.

- Many users would doubtless prefer to opt out of a network and "do their own thing" without corporate interfaces (and expenses).

- This problem may solve itself in the long run, as individual department DSS efforts require data collected throughout the company.

- A fact of life that IS management will have to accept is that DSS-generated code will not be as efficient as that in traditional systems.

  - Happily, decreasing hardware prices will mask much of this. (See INPUT's 1981 Vendor Watch Report, New Storage Systems and Their Implications.)

- Initially, there will be much confusion over ownership, use, and meaning of data. Data base builders will struggle to keep up (and will sometimes lose). This could be very dangerous from a corporate standpoint.

  - Departments could prosper at the expense of the company. This is, of course, impossible to overlook for long.

  - Critical corporate data resources must be guarded carefully.

- DSS activity can lead to explosive surges in hardware use as a particular DSS application exercises the corporate data base.

  - As long as the activity is useful, it will be hard to say no. Effective capacity planning strategies and tactics are a must. (See INPUT's 1981 Impact Report, Performance Measurement And Capacity Planning.)

  - The most serious problem and the one that will be most difficult to deal with is the potential for security breaches in a DSS environment.

  - The following combination of events could be explosive:

. Systems built by unsophisticated users.

. Few operational safeguards.

. Temporary systems that become permanent.

. No documentation.

. High value/high visibility system.

. Large amounts of money involved.

- The final chapter provides recommendations on how to deal with these problems.

- It will help to deal with the negative factors if it is appreciated how they flow directly from one or more positive factors, as shown in Exhibit IV-1.

- Curbing some of the excess exuberance can help to limit the potential for damage inherent in decision support systems.

B. DSS GROWTH FACTORS

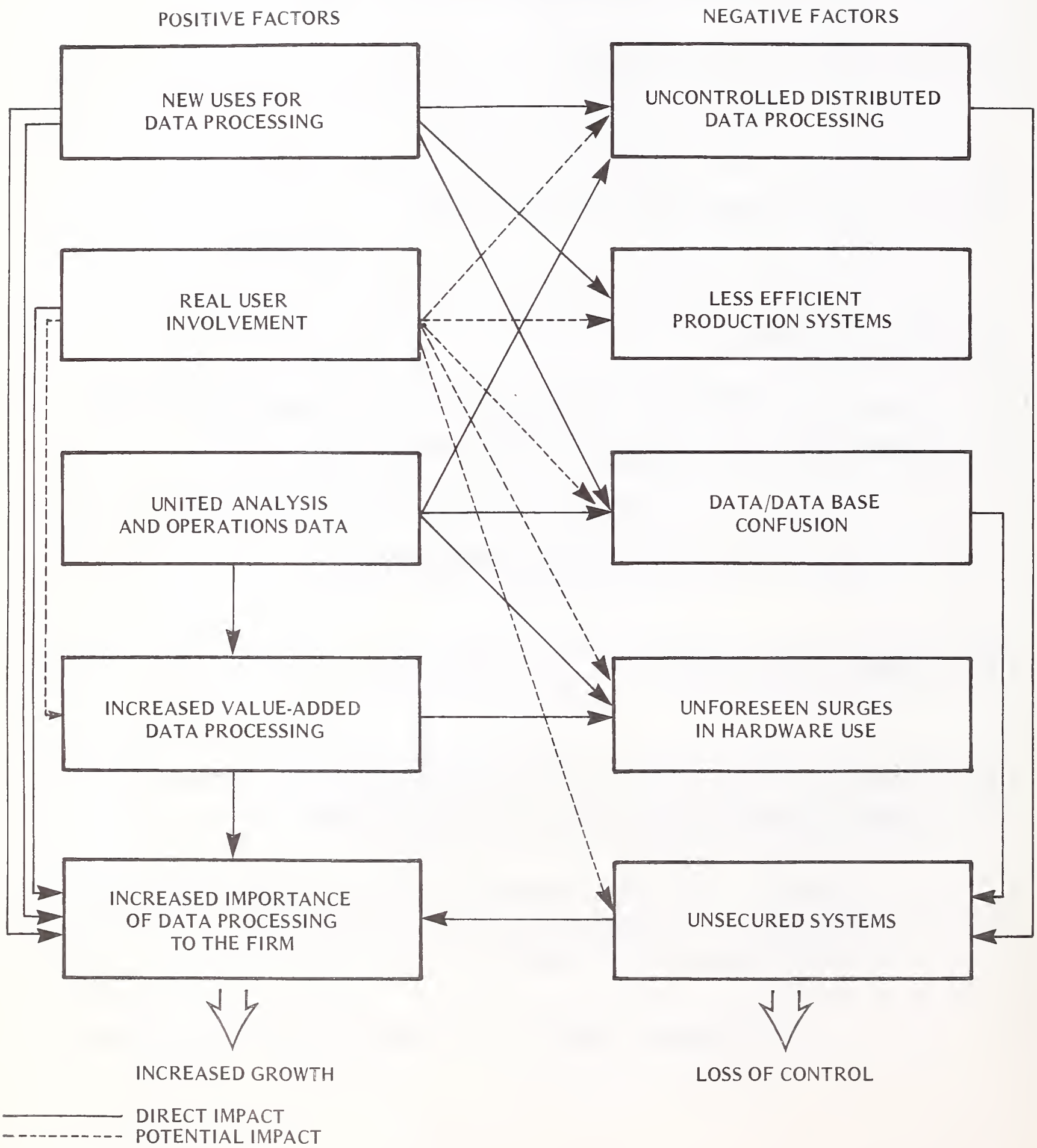
- The use of decision support systems should see a dramatic increase in coming years. This growth will be fueled by the following factors.

- Planning - Even though forecasting, etc. have not been tremendously successful so far, there are very strong pressures for more planning, given current economic uncertainties.

. No company which INPUT surveyed foresaw less planning activity.

EXHIBIT IV-1

THE TWO FACES OF DSS



. It is human nature to try to do something about a problem, even if it does not solve the problem.

- Example - There are a great many DSS success stories and these set an example.

. People do not want to be the "last kid on the block" to get the new toy.

. Already decision support systems are used in an impressive number of areas. Exhibit IV-2 shows typical DSS uses in only one area, financial modeling.

- Marketing - An increasingly strong force is the marketing effort of DSS software and timesharing firms.

- Tools - Management science approaches and tools are increasingly sophisticated and accepted. They often produce identifiable results.

. Hardware to support them is getting cheaper and easier to use.

. A wider choice of software is available. This is a key factor.

- Ease of Use - This is linked to software and other tools, but really stands out on its own.

. Successful decision support systems are based on really user-friendly systems.

. The explosive growth in "convenience" copiers arose after copiers were no longer inconvenient to use.

EXHIBIT IV-2

FINANCIAL MODELING: TYPICAL USES

- Product planning
- Pricing strategy
- Investment analysis
- Financial statement consolidation
- Joint venture reporting and analysis
- Long-term financial planning
- Acquisition analysis
- Capital budgeting
- Lease versus purchase
- Cash management
- P & L forecasting
- Risk analysis

C. THE POTENTIAL FOR DSS FAILURE

- Like most human enterprises, DSS practitioners and theorists focus to a large degree on decision support systems that are working and used - the hits. Many decision support systems "close out of town." The reasons for this are varied but fall into two general classes:

  - Those caused by technical problems.

  - Those caused by a lack of acceptance.

1. TECHNICAL FAILURES

- The technical issues are many and varied, but for the most part will be recognizable by those who have been in data processing for a while; a partial list is given in Exhibit IV-3.

  - Many of these problems are caused by the nature of the problem/solution not being sufficiently understood before plunging in.

  - In many ways, this is a strength of a DSS. The theory behind it and much DSS software assume it to be the case.

    . Happily, the insuperable problems are almost always identified before too much in the way of time, resources and, especially, promises have been committed.

    . Contrast this to classical systems where it may be literally years before the truth has sunk in that a much-touted new system simply will not do what was counted on. This can be learned in a matter of days in many DSS implementations.

EXHIBIT IV-3

EXAMPLES OF DSS FAILURE CAUSED BY TECHNICAL PROBLEMS

- Software unsuited to job
  - Original selection wrong
  - Job evolved beyond original expectation
- Objectives too vague or unstructured to be well quantified
  - Often not known until it is tried
- Inadequate/insufficient technical support
- Data unavailable and/or not understood
- Data come "unstuck" and/or out of synch
  - Especially a problem where data are marshalled from many sources for time series analysis
- Attempt to put system into regular production, that is unsuited technically
  - A special danger for decision support systems
- Run costs too high for benefits
  - Where an outside timesharing service is used
- Takes too many hardware/people resources supplied by others
  - For internally developed systems

- Note that for the most part DSS failures are not the failures of the classic, large DP system, i.e.:

  . Too late.

  . Over budget.

  . Do not meet user needs.

2. LACK OF ACCEPTANCE

- Failures caused by a lack of direction or support are almost unique to decision support systems. While the system may work in a technical sense, it really does not accomplish anything: the operation was a success but the patient died.

  - According to a leading DSS software vendor, while ongoing top management involvement is critical for success, "plenty do not stay involved." Planners and analysts too often plan in isolation and are out of touch with changing top management needs.

- One of the biggest weaknesses in decision support systems (one not often talked about much in public) is their lack of credibility in some management circles.

  - Obviously, there are some seat of the pants managers who couldn't give two cents for planning in general, and computerized planning in particular.

  - The more thoughtful managers have doubts much more difficult to deal with: How can we plan and forecast within our company when so much is dependent on ill-understood external financial and economic variables?

  - According to this position, it is not enough to say that a certain project is very sensitive to a high inflation rate if the planners cannot come up with a credible inflation rate scenario.

- Many of the financial planners that INPUT spoke to were frankly defeatist in their ability to really forecast the future. There have been too many financial shocks and turning points which no one had foreseen.

  - As L.B. Mayer once put it, "The trouble with forecasting is that it's so hard to tell what's going to happen in the future."

  - A representative of a DSS vendor whom INPUT interviewed said that many users of their package were frankly "frustrated by having good ideas that were not appreciated within the company." In large part this was a direct result of the unsuccessful efforts of "academic economists to forecast general economic movements accurately."

- One of the most surprising problems associated with DSS is that companies often do not validate the results of their forecasts over time. (Most decision support systems, explicitly or implicity, are making statements concerning what would be happening in the future.)

  - A leading vendor of DSS software said that the fact that "many companies do not perform routine post-mortems on their forecasts affected DSS credibility."

  - This was amply confirmed in a series of interviews which INPUT conducted with financial planners.

    - Many had obviously never even considered reviewing actual against planned performance. Several, in the course of the interview, thanked INPUT for the suggestion.

. Others said that it had been considered, or conducted in a cursory fashion, but that it had not been done in depth because they "knew" that later events (primarily inflation) had invalidated their forecasts!

. It should be noted that the planners interviewed were not in companies noted for their planning efforts and achievements. However, they were all firms well up on the list of "Fortune 500" companies and by all evidence are representative of those companies that do try to plan.

- Sometimes there can be a DSS failure caused by its doing its job too well. The result is correct but it goes so strongly against entrenched folklore that it is not accepted.

  - This is a failure of presentation and, also, one caused by not involving the real decision-makers early enough.

  - This problem is often caused by bright staff people trying to impress their supervisors with what "I" can do, rather than what "we" can do.

V CONCLUSIONS

V CONCLUSIONS

A. FINDINGS

- This report has taken the following positions:

  - The activities that now fall under the general heading, "Decision Support Systems," are one of the main growth areas open to the IS department (the others being office automation and distributed processing).

    . DSS will, however, be the most visible to top management and may have a disproportionate effect on the credibility of the IS department.

  - The IS department can have little, if any, effect on which DSS projects are selected or on the priority and timing of their implementation.

    . The forces which drive DSS activities are located deep within the business itself - an area to which most IS managers are not invited or do not wish to go.

    . To make DSS projects conform to the classic DP life cycle, planning and implementation will be at best futile and at worst successful (i.e., they won't work, or will not be timely).

- Furthermore, IS departments will take a leading role in DSS development at their own risk.

  . Decision support system development only makes explicit what many implementors of traditional data processing systems regularly discover: users do not know what they want until they have seen what they do not want.

  . The high visibility, lack of complete definition, and time pressure in the DSS environment make it critical that the most senior user/decision-maker possible have hands-on <u>functional</u> (not technical) control over the DSS's development.

  . Imposing IS standards or personnel is a sure road to failure.

  . User departments will squirm out of IS control, one way or another; they will often turn to a timesharing service (perhaps calling it "consulting" for expenditure purposes).

- Does this mean, then, that IS departments will tend the waterworks while the DSS departments set up wineries? Not necessarily, but the IS department will have to adapt to new ways.

B. THE CHANGING ROLE OF INFORMATION SYSTEMS

- The IS department will have to change from a group that tries to accomplish its ends by direct action to one that achieves its objectives indirectly in the areas shown in Exhibit V-1.

  - Applications.

  - Systems analysis and development.

EXHIBIT V-1

THE CHANGING INFORMATION SYSTEM DEPARTMENT ROLE

ACTIVITY AREA	IS DEPARTMENT ROLE	
	OLD ROLE	NEW ROLE
Applications (i.e., data and systems)	Control	Suggest
Systems analysis and development	Implement	Advise
Technical knowledge	Perform	Support
Procedural knowledge (e.g., controls, documentation)	Specify	Recommend

- Technical knowledge.

- Procedural knowledge.

- Applications will no longer be largely controlled by the IS department; rather, IS staff, based upon their perceived experience will suggest to DSS developers the kind of data that will be required to fulfill the objective the developers have in mind.

  - A very important role, requiring much tact and knowledge, will be that of a "data broker."

  - The data broker function will be to describe and interpret data needed by the DSS but unfamiliar to the DSS builder/department. Both the technical support of a data dictionary as well as a knowledge of the business meaning of the data involved will be critical to the success of this function.

- Systems development will be largely undertaken by "system illiterates," in the sense that few user department people will be DP professionals. However, with the proper user friendly tools the DSS consumer can get what is actually needed, rather than a complex, sophisticated product that is somewhat off target.

  - As indicated earlier (see Exhibit III-1) the basic activities in constructing a "decision support system" are very similar to those performed in developing any system.

  - The IS department should be able to share its accumulated experience with user departments, many of which will always be in the neophyte stage, owing to employee turnover.

- Currently, most technical knowledge (e.g., programming techniques, software internals, package characteristics, etc.) is used within and for IS department

operations. DSS users will have a great need for technical information; they will be aware of a significant portion of their needs in such areas as:

- Characteristics of and differences between DSS software alternatives, including their strengths and weaknesses.

  . There are consulting organizations which specialize in this type of analysis; the extent to which the IS department can or should become involved in the selection process will depend on its resources and the importance of the acquisition to the company.

- How to use the DSS software (initially) and, then, how to get the most out of it.

  . Where there is a vendor involved, a certain amount of training and handholding is available at the time of acquisition.

  . However, user departments often have a recurring need (because of turnover) for basic training and a constant need for brushing up (because of the lack of intensity of use).

  . While these needs can be satisfied by vendors, it is often more satisfactory (because of personnel availability and company-based knowledge) and generally cheaper to have such training conducted in-house.

  . However, there has to be a large enough critical mass of DSS users to make it worthwhile to set up an in-house support function. If an in-house support function is established, it must take its job seriously, since DSS users will always need their help now.

- One of the strengths of the classic system life cycle approach is its attention to procedural requirements, i.e., accounting and security controls, documen-

tation, test protocols, program structure, etc. These are a struggle to do well in even the best run organizations since the work is not glamorous and does not contribute anything obvious to initial project "success."

- However, most IS departments have learned the hard way that all of these measures are required for longer term systems success.

- Many of these traditional requirements are not deemed necessary in many one-time DSS applications:

  . The developers understand what they are doing and do not intend to do it again, consequently, written documentation is deemed superfluous.

  . Efficient or even well-written programs are similarly not required.

  . Formal testing may not be needed because the users "know" the data.

- Actually, most procedural safeguards are ignored because they are never even considered (or, perhaps, known about).

- In fact, many procedural safeguards <u>are</u> needed, even in a supposedly DSS environment.

  . Testing protocols will always be needed to ensure that the program or model is working as intended. Somehow, one's objectivity regarding data greatly diminishes when it is one's own child that is performing.

  . Similarly, some form of documentation may be required when, for example, months after a model is supposedly finished there are requests for variations (from the Board of Directors, say). It

then becomes very important to be able to go back to, perhaps, the sixth iteration of the base case and make further sensitivity analyses.

. The same argument supports a structured approach to coding so that those who were not party to the creation of the original decision support system can step in and take over the system.

. Security (and controls in general) is usually overlooked in user-controlled DSS development. Controls are very important to ensure that all of the right data and none of the wrong data are being used. Physical security is not usually an issue since little of the data are irreplaceable. However these data are often very valuable and sensitive; is there adequate protection?

. The procedural overkill often associated with large system development is definitely not needed. However, a focused version consistent with a DSS environment is certainly in order and should be developed.

. To have such aids actually used, though, will require a selling job to the user departments. The best approach is the one used so effectively by CPA firms - a workable solution accompanied with a stern warning that if any trouble arises, the blood is on the user's hands. This works well in large, essentially bureaucratic, organizations.

- The IS department in conjunction with the internal audit staff should establish a DSS auditing and supervision function both to offer user DSS developers control tools and to ensure that the core controls are used - and used correctly.

- For the most part, though, the IS department should be viewed as a friend and supporter and not a policeman. This will require a number of attributes that are now fairly unusual in IS departments:

  - IS should stay in the background. If the user department is inclined to take all the credit for good ideas, let them. Other potential users of IS services will realize what is happening and will feel much more at ease in looking for the support of the IS department.

    . Few consultants or vendors look for explicit credit within an organization - that is what brings them more business.

  - It may be a truism that good service has to be supplied consistently, but this is often not followed by IS departments that are supporting internal clients.

    . Good people are borrowed (stolen) for "important" IS development projects.

    . In-house timesharing systems sometimes give erratic responses because of other priorities.

  - Adequate hardware resources must be maintained so that all reasonable (as well as a certain number of unreasonable) storage and response time needs can be met.

    . Often, the only way to ensure this is to allow decision support systems to be run on an outside timesharing service or on a user department micro or mini.

    . Sometimes, of course, all a department may really need is VISICALC on an Apple. It is the responsibility of the IS department to know that and so advise the user.

- Sometimes a DSS has been so successful and useful that the user department wants to use the system on an ongoing basis. This is known as going into production by the back door.

  - This should become more common as users realize they can jump to the head of a, perhaps, two-year waiting line for medium priority jobs and get the system they really want in the bargain.

- It makes some difference if the user department is willing to actually run the system itself. At least that burden is off the IS department.

  - Actually, the user department usually cannot run the system, at least not at a better level of performance than was common at the dawn of the 360 era. This is because a decision support system can often be quite complex: data coming from a number of sources in a number of forms with the sequence and timing often critical. Quite soon the user will be very unhappy, perhaps suffering a disaster. DP operations may find themselves saddled with a sickly orphan and no chance to say no.

- It is this situation that makes it so difficult for the IS department to take the position that: "Here are our standards: meet them and we'll run your jobs."

- Another option is to run jobs on an outside timesharing service. However, this is usually very expensive for production work and is, in any event, embarrassing for an IS department that has won its spurs by bringing timesharing in-house.

- Even if a DSS is a clean system that is easy to run it will almost certainly be very inefficient.

  - This is the tradeoff made by user-friendly systems.

  - "If only they were written in good, efficient COBOL." (Heresy even ten years ago, but a reality now.)

- There is probably a need for a DSS to COBOL converter (or a DSS statement use analyzer), but none yet exist. In fact the trend is in the other direction with talk of using third-generation DSS packages to write fourth-generation packages.

- A way out of this dilemma is to view the decision support system as a form of systems analysis. When the user declares the system ready for production what is really being said is that the system is ready for coding (and perhaps design also).

  - This will often be an extreme step, in view of constantly falling processing and storage costs and the expense and scarcity of programmers. But this way IS management can make a rational choice between hardware costs and people costs.

  - These costs should be brought home to the user by, for example, costing out user-site hardware assuming efficient and inefficient code.